(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
17 November 2005 (17.11.2005)

PCT

(10) International Publication Number
WO 2005/109824 A1

(51) International Patent Classification[7]: H04L 29/06

(21) International Application Number:
PCT/US2005/011702

(22) International Filing Date: 5 April 2005 (05.04.2005)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
10/832,588 27 April 2004 (27.04.2004) US

(71) Applicant *(for all designated States except US)*: **CISCO TECHNOLOGY, INC.** [US/US]; 170 West Tasman Drive, San Jose, CA 95134-1706 (US).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: **ITHAL, Ravishankar, Ganesh** [IN/US]; 1401 Red Hawk Circle, Apartment E106, Fremont, CA 94538 (US).

(74) Agents: **HUANG, David, E.** et al.; Chapin & Huang, LLC, Westborough Office Park, 1700 West Park Drive, Westborough, MA 01581 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

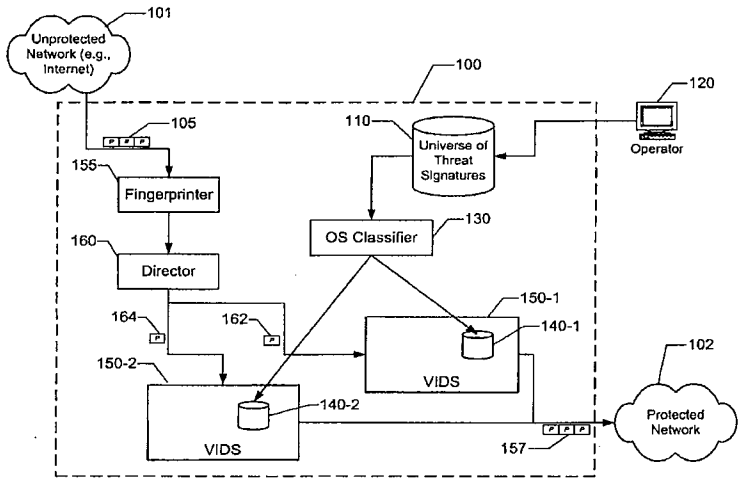
(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii)) for the following designations AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NI, NO, NZ, OM,*

[Continued on next page]

(54) Title: SOURCE/DESTINATION OPERATING SYSTEM TYPE-BASED IDS VIRTUALIZATION



(57) Abstract: Systems and methods for virtualizing network intrusion detection system (IDS) functions based on each packet's source and/or destination host computer operating system (OS) type and characteristics are described. Virtualization is accomplished by fingerprinting each packet to determine the packet's target OS and then vetting each packet in a virtual IDS against a reduced set of threat signatures specific to the target OS. Each virtual IDS, whether operating on a separate computer or operating as a logically distinct process or separate thread running on a single computer processor, may also operate in parallel with other virtual IDS processes. IDS processing efficiency and speed are greatly increased by the fact that a much smaller subset of threat signature universe is used for each OS-specific packet threat vetting operation.

WO 2005/109824 A1

PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, UZ, VC, VN, YU, ZA, ZM, ZW, ARIPO patent (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG)

— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii)) for all designations*

Published:

— *with international search report*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

SOURCE/DESTINATION OPERATING SYSTEM TYPE-BASED IDS VIRTUALIZATION

BACKGROUND

A typical computer networking system may include, among other things, an intrusion detection system (IDS) configured to monitor network traffic and to block attempted attacks on or intrusions into the protected network space. Such intrusion detection systems may include or coexist with various types of firewalls, packet monitors, and other devices that typically include intrusion sensing functions (e.g., advanced routers). These systems include both active and passive devices and are generally referred to as "sensors."

An IDS may include, among other things, a network interface for receiving packets, a packet filtering mechanism for determining whether or not to accept inbound packets, memory for storing threat signatures, and a network interface for transmitting (or forwarding) packets into the protected network. The aforementioned elements of an IDS may be implemented in either hardware or software or some combination of both.

An IDS sensor may also be virtualized. "Virtualized," as this term is used in the art, refers to virtualization, the practice of dividing the IDS functionality among multiple processes or logical elements each configured to operate in parallel with the others. These processes may run on separate processors (i.e., in separate pieces of hardware) or may run on a single processor in multiple threads. In this sense, virtualization may be thought of as another form of distributed processing: the functional or logical elements of the required process may be carried out in multiple locations, where "location" is understood as referring to both physical as well as logical separation. One of ordinary skill in the art will also recognize that virtualization does not necessarily require the division of IDS functionality into separate threads; the "feel" of providing multiple logical functions within a single physical device is all that is needed.

One conventional approach to IDS virtualization is to use to separate functionality (or to "virtualize") based on user-configured selection criteria such as packet IP prefix, domain name, VLAN, input interface, etc. In order to provide the same level of threat protection in each virtual IDS (or virtual IDS process), however, packets destined to be processed by a given virtual IDS sensor must be checked against signatures of vulnerabilities for all

operating systems known to exist on the unprotected network. For typical cases where the unprotected network consists of the public Internet, this universe of operating systems is the universe of all known operating systems. Likewise, when the protected network is sufficiently large and diverse, the vulnerabilities in the destination host must include the vulnerabilities of all known operating systems as well.

In typical IDS systems, known vulnerabilities are stored in IDS memory as threat signatures, i.e., descriptive information formatted so that it may be rapidly compared by the IDS to packet content in order to directly determined whether each particular threat is or is not present in the packet. As the number of threats grows, so too must the set or universe of threat signatures. As each new threat is identified, any aspect or manifestation of that threat not common to a previously seen threat signature necessitates the definition of a new threat signature.

SUMMARY

There are several notable deficiencies to the above-described conventional approaches. For example, the prior art systems do not generally scale well as the number of threat signatures continues to increase. In addition, the packet latency introduced by comparing (or "vetting") each incoming packet against the universe of threat signatures causes a significant degradation in performance. In general, the vetting of each packet against each threat signature in the signature universe is slower and more inefficient than is desirable in modern high-speed, high throughput packet processing IDS sensors.

In contrast to the above-described conventional approaches, embodiments of the invention are directed to systems and methods for virtualizing IDS functions based on the operating system (OS) characteristics of each packet's source and/or destination host computers. This virtualization is accomplished by fingerprinting each packet to determine the packet's target OS and then performing a rapid packet vetting against a reduced set of threat signatures appropriate to that target OS. The "target OS" may be either the operating system of the packet's source host or the operating system of the packet's destination host. In some embodiments, information about either or both OSs may be used to select a reduced set of threat signatures.

The vetting process for each packet proceeds by comparing aspects of the packet such as, but not limited to, packet data payload, header flags, options, source IP address destination IP address, source port and/or destination port to the reduced threat signature set appropriate to the target OS for each particular packet.

Accordingly, each virtual IDS process, whether operating on a separate machine (computer), operating as a distinct thread running on a single computer or processor, or merely a logical distinction in functionality is thus able to operate in parallel with other virtual IDS processes, although parallel operation is not necessary to practice the present invention. In such an embodiment, each VIDS process performs packet vetting operations using the reduced threat signature set appropriate to the target OS as determined by the contents of each packet. IDS processing efficiency is greatly increased by the fact that a much smaller subset of threat signature universe is used for each packet threat vetting operation. Furthermore, as the universe of threat signatures increases without bound (as it is currently expected to do), IDS efficiency will not be greatly impacted since packets not subject to new vulnerabilities will not have to be vetted against those new threats. In fact, processing efficiency may be greatly enhanced by the ability of embodiments of the invention to greatly reduce the set of threat signatures that need to be searched for each packet. For example, it is well-known in the art that Apple Computer's OS X operating system is not vulnerable to the vast majority of attacks seen "in the wild" (i.e., known to be loose) today. Packets destined for this OS thus do not need to be vetted against threats to the Microsoft Windows OS, for example. Since the number of threat signatures is so much smaller, processing (including threat vetting) speed for OS X-destined packets is greatly increased.

BRIEF DESCRIPTION OF THE DRAWINGS

The foregoing and other objects, features and advantages of the invention will be apparent from the following description of particular embodiments of the invention, as illustrated in the accompanying drawings in which like reference characters refer to the same parts throughout the different views. The drawings are not necessarily to scale, emphasis instead being placed upon illustrating the principles of the invention.

Figure 1 is a high-level block diagram of an intrusion detection system configured according to one embodiment of the present invention.

Figure 2 is a high-level flowchart of the process of IDS virtualization, according to one embodiment of the present invention.

DETAILED DESCRIPTION

Embodiments of the present system are directed to techniques for intrusion detection system (IDS) virtualization based on packet target operating system (OS) characteristics and the tailoring of packet vetting to a reduced threat signature set. Tailoring, and the concomitant IDS virtualization, is based in turn on the use of passive and/or active packet fingerprinting to determine the packet's target operating system. The target OS may be, in some embodiments, a tuple consisting of the operating system of the packet's source host and the packet's destination host.

Figure 1 illustrates a virtualized intrusion detection system 100 configured according to one embodiment of the present invention. Viewed at a high level, system 100 consists of an interface to an unprotected network 101, the vetting/processing system components, and a second interface to a protected network 102. Additionally, an operator interface 120 allows an operator to load the universe of threat signatures.

Within the vetting/processing components of system 100, packets are processed by a fingerprinter, which in turn supplies fingerprinted packets to a director for switching or routing to one or more virtual IDS (VIDS) processes (which, as noted above, may be logically distinct in any of several ways). After vetting (with the aid of reduced threat signature sets 140) in the appropriate VIDS process (or set of VIDS processes), packets leave system 100 via the second network interface and enter protected network 102.

In one embodiment of the invention in particular, a stream of packets 105 from an unprotected network (such as, but not limited to, the public Internet) enters system 100 at fingerprinter 155 via a first network interface (not shown). Fingerprinter 155 looks at each packet to determine both the host operating system that sent the packet as well as the operating system of the host computer for which the packet is destined. This fingerprinting process, carried out on each packet in packet stream 105, is accomplished using either passive or active fingerprinting methods and techniques commonly used and well-known in the art. Passive fingerprinting techniques are discussed in detail in Toby Miller, Passive OS Fingerprinting: Details and Techniques, available at http://www.sans.org/rr/-

special/passiveos.php (last viewed 4/13/04), and Passive OS Fingerprinting: Details and Techniques (Part 2), available at http://www.sans.org/rr/special/passiveos2.php (last viewed 4/13/04), both incorporated herein by reference in their entireties. Active fingerprinting techniques are discussed in detail in (for example) Fyodor, Remote OS detection via TCP/IP Stack Fingerprinting, available at http://www.insecure.org/nmap/nmap-fingerprinting-article.html (last viewed 4/13/04), and Ofir Arkin, ICMP Usage In Scanning, Version 3.0, available at http://www.sys-security.com/archive/papers/ICMP_Scanning_v3.0.pdf (last viewed 4/13/04), both of which are incorporated herein by reference in their entireties. As the design and configuration of the hardware and/or software necessary to carry out either passive or active fingerprinting (or both) is well within the skill of an ordinary practitioner, implementation of specific fingerprinting techniques is not further discussed herein.

As each packet is fingerprinted, information denoting the source and destination operating systems is associated with the packet. This information may be stored, for example, in a temporary or scratchpad memory for use by director 160. Alternatively, target operating system data may be appended to or concatenated with the packet as it is passed to director 160.

In many implementations, the destination operating system (i.e., the operating system of the destination host) is of most use in evaluating the threat posed by packet; the "target OS" is the destination host OS because the most common threat is a threat to the packet's destination. In some cases, however, certain threat signatures are defined in terms of the source operating system, i.e., the "target OS" is the operating system of the host that sent the packet. For example, if the source operating system can be identified through fingerprinting as UNIX, then the fact that the packet contains a Microsoft Windows remote procedure call (RPC) indicates that a UNIX box may be attempting to penetrate a Windows system. Since a Windows RPC call would not normally come from a UNIX machine, it should be identified as a threat. Thus, some signatures may reference the source host OS as the target operating system.

Packets leaving fingerprinter 155 enter director 160 where, based on the target OS identified by fingerprinter 155, they are sent (i.e., redirected, switched, or routed, as those terms are known the art) to the appropriate virtual IDS (VIDS) process 150.

Although fingerprinter 155 and director 160 are depicted in Fig. 1 as separate functional elements, one of ordinary skill in the art will appreciate that these functions may be implemented in one or more devices and/or one or more to software processes. Accordingly, the implementation of virtualized IDS 100 is not limited to a particular distinction between or arrangement of hardware and/or software functionality and physical embodiment.

For the sake of clarity, Fig. 1 illustrates only two VIDS processes, 150-1 and 150-2, although in a typical implementation there may be many VIDS processes or units, each corresponding to a different target operating system. And, although each VIDS 150 is described as a process executing in software on a processor circuit, those skilled in the art will realize that the functions of a VIDS process may also be implemented in hardware (i.e., in a processor or other computer unit), in a single software process, or in a combination of hardware and software. Accordingly, the present invention is not limited to any particular implementation of the VIDS functionality.

By way of illustration, but not of limitation, VIDS 150-1 may be configured (in some embodiments of the invention) to compare individual packets 162 to a particular reduced threat signature set 140-1, corresponding to threats targeted at (for example) Windows XP. A second VIDS, 150-2, may be configured to compare packets 164 (which are directed to VIDS 150-2 because they share a target operating system that is different from that shared by packets 162) to a second reduced threat signature set 140-2. Reduced threat signature set 140-2 may list threats targeted at (for example) Windows 98. One of ordinary skill of the art will appreciate that many VIDS units 150 may be employed, each containing a reduced threat signature set 140 containing threat signatures appropriate to a different target operating system. Accordingly, the present invention is not limited to a particular number of VIDS units 150 and/or reduced threat signature sets 140.

Reduced threat signature sets 140 are formed by OS classifier 130 from a database or other memory or storage device 110 containing the universe of all known threat signatures. The universe of threat signatures may be loaded into database 110 prior to activation of the virtualized IDS 100 by an operator using a conventional workstation or computer 120. Alternatively, although not shown, the universe of threat signatures 110 may be loaded by file transfer, scan, self-discovery or monitoring of network traffic, or any of the conventional

means known in the art or yet to be discovered for creating and maintaining a database of threat signatures.

The process of forming discrete, reduced threat signature sets 140 from the universe of threat signatures 110 is accomplished by selecting signatures according to target operating system through conventional sorting methods and techniques. The threat signatures themselves are conventional text and/or digital data strings known and used in the art for packet-based threat vetting. The use and organization of such threat signatures are described in, for example, Kyle Haugsness, Intrusion Detection In Depth: GCIA Practical Assignment Version 3.0, (December 2, 2001), available at http://www.sans.org/rr/papers/23/835.pdf (last viewed 4/13/04); Syed Yasir Abbas, Introducing Multi Threaded Solution to Enhance the Efficiency of Snort, (December 7, 2002), available at http://www.cs.fsu.edu/-research/reports/TR-021204.pdf (last viewed 4/14/04); and the Snort Users Manual, v. 2.1.2, available at http://www.snort.org/docs/snort_manual/, last viewed 4/23/04), all of which are incorporated herein by reference in their entireties.

The actions of operator 120 and OS classifier 130 required to load threat universe database 110 and to select or form the various reduced threat signature sets 140 may be accomplished at any time prior to activation of virtualized IDS 100 or even during operation, as when certain threat signatures must be updated without interrupting packet processing in virtualized IDS 100. Accordingly, although reduced threat signature sets 140 must be initially defined, those definitions need not be static and the present invention is not so limited.

Certain threats are known to be common to more than one target OS or platform. Accordingly, threat signatures representing these common threats are grouped into a separate reduced threat signature set 140 (not shown) that is necessarily applied (in another VIDS 150) to all packets 105. This parallel processing capability may be implemented (in some embodiments of the invention) in director 160 by providing the capability to direct packets to more than one VIDS 150 at the same (or substantially the same) time. This capability for parallel (near-simultaneous) processing of packets by more than one VIDS 150 provides a significant speed and throughput increase in virtualized IDS 100.

Packets 162 or 164 (in the two-VIDS 150 embodiment of Fig. 1) that match a threat signature are blocked within VIDS 150 and not allowed to pass through into protected

network 102. Blocking may take any conventional form, such as but not limited to deletion, routing to /dev/null, packet marking and routing to a special process for alarming/reporting, or some other means of preventing the threatening packet from leaving virtualized IDS 100. Conversely, packets that do not match any threat signature form a vetted (or accepted) packet stream 157, which is then passed on (through conventional means) into protected network 102. Protected network 102 may consist of, for example but not by way of limitation, an intranet, LAN, MAN, or other relatively-closed network space secured and protected by virtualized IDS 100.

Figure 2 illustrates one exemplary embodiment of a process 200 whereby the virtualized IDS receives packets, performs fingerprinting and packet redirection to an operating system-appropriate VIDS process, compares each packet to a specific reduced threat signature set, and passes or rejects packets accordingly.

Process 200 begins prior to virtualized IDS packet processing operations with the loading of threat signatures in step 210. Step 210 defines (or collects) the universe of relevant threat signatures in a database or other memory or storage element. The process of signature definition and/or collection (including editing and/or revising as required) may be performed, in some embodiments of the invention, by conventional techniques and processes well-known in the art. Process 200 next forms from the threat signature universe a number of reduced threat signature sets, based on target operating system, in step 220. As noted above, each threat signature is specific to a target operating system, i.e., the threats themselves are generally targeted at only a single operating system. For threat signatures that apply to multiple operating systems (for example, vulnerabilities that exist in more than one operating system), a reduced threat signature set is formed for threats common to two or more operating systems. Each reduced set of threat signatures is stored in a database or other memory structure using conventional processes.

Each database (or set) of operating system-specific reduced threat signatures is then used, step 230, to form a virtualized IDS process configured to compare or vet each packet presented to it. These packets begin to arrive at comparing step 280 once the virtualized IDS begins to process incoming packets in step 250.

The virtualized IDS receives packets 250 and fingerprints each one in step 260. Fingerprinting 260 may be an active process, such as by querying a DHCP server or by other

active fingerprinting methods known in the art. Alternatively, fingerprinting 260 may be accomplished by passive means, such as (but not limited to) an analysis of the packet header's ACK, Flags, and/or Options fields and comparison of the values thereof to a set of OS-specific indicators. These indicators (or rules) permit the inference of the packet source host's and/or the packet destination host's operating system from the TCP and/or IP packet header field values.

After fingerprinting step 260 has determined the identity of the target OS for each packet, the packet is directed (in step 270) to a particular one of the several VIDS processes according to the packet's target OS. As discussed above, multiple VIDS processes may be created (in step 230) to match the number of distinct reduced threat signature sets formed in step 220.

In some implementations (not shown in Fig. 2), the process 200 checks whether the protocol carried by each packet can be officially "talked" (used, communicated, sent) by the source OS by further directing the packet (in step 270) to a second VIDS process for evaluating the source OS. The source OS VIDS checks only if the protocol is valid for the source OS, as further described below. A valid protocol passes the VIDS comparison for the source OS.

Within each VIDS process 275, each input packet is compared, step 280, to the corresponding reduced threat signature set for the packet's target OS. If the packet passes the comparison test, i.e., the packet does not match any threat signature in the reduced threat signature set, it is accepted for further processing and/or forwarding out of the VIDS process 275 in step 290. The VIDS process 275 then continues or loops back to comparison step 280 for the next packet presented.

If, on the other hand, the packet matches one of the threat signatures in the reduced threat signature set, the packet is dropped in step 299. "Dropped," in this context, refers to any of the various ways in which a bad packet may be processed in the intrusion detection context: the system may alternatively set an alarm, flag, reject, or null-route the packet, or otherwise suppresses its further transmission from the virtualized IDS.

It must be noted, although not depicted in Fig. 2 due to the need for clarity and simplicity in the drawing, multiple VIDS processes 275 (as represented by step 280, 290 and

299) may operate in parallel in some embodiments of the invention, thereby processing multiple packets at substantially the same time. In addition, more than one VIDS process 275 may process the same packet, as when both an OS-specific threat signature set and a common threat signature set are to be applied. Regardless of whether parallel processing is employed, however, the reduction in the threat signature set applied to each packet provides a significant performance benefit.

Alternate Embodiments

The order in which the steps of the present method are performed is purely illustrative in nature. In fact, the steps can be performed in any order or in parallel, unless otherwise indicated by the present disclosure.

The method of the present invention may be performed in hardware, software, or any combination thereof, as those terms are currently known in the art. In particular, the present method may be carried out by software, firmware, or microcode operating on a computer or computers of any type. Additionally, software embodying the present invention may comprise computer instructions in any form (e.g., source code, object code, interpreted code, etc.) stored in any computer-readable medium (e.g., ROM, RAM, magnetic media, punched tape or card, compact disc (CD) in any form, DVD, etc.). Furthermore, such software may also be in the form of a computer data signal embodied in a carrier wave, such as that found within the well-known Web pages transferred among devices connected to the Internet. Accordingly, the present invention is not limited to any particular platform, unless specifically stated otherwise in the present disclosure.

While this invention has been particularly shown and described with references to embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the spirit and scope of the invention as defined by the appended claims.

CLAIMS

I claim:

1. A method of intrusion detection system virtualization, comprising:
receiving a stream of packets;
fingerprinting each packet in a said stream to identify at least one target operating system (OS) type;
directing each said packet to a virtual IDS process corresponding to each said identified target OS type;
comparing each said packet to a threat signature set corresponding to each said identified target OS type in said virtual IDS process; and
accepting each said packet based on said comparing.

2. The method of Claim 1, wherein said fingerprinting comprises active fingerprinting.

3. The method of Claim 1, wherein said fingerprinting comprises passive fingerprinting.

4. The method of Claim 1, wherein said target OS type is the packet source host OS.

5. The method of Claim 1, wherein said target OS type is the packet destination host OS.

6. The method of Claim 1, wherein said target OS type is both the packet source host OS and the packet destination host OS.

7. The method of Claim 1, wherein, when said fingerprinting identifies more than one target OS type, said directing occurs at substantially the same time.

8. A method of intrusion detection system virtualization using operating system type information, comprising:
forming a plurality of reduced threat signature sets from a signature universe based on operating system type;
virtualizing an intrusion detection system (IDS) into a plurality of virtual IDS processes corresponding to said plurality of reduced threat signature sets;

receiving a stream of packets;

fingerprinting each packet in a said stream to identify at least one target operating system type;

directing each said packet to the virtual IDS process corresponding to each said target operating system type;

comparing each said packet to said reduced threat signature set in said virtual IDS process; and

accepting each said packet based on said comparing.

9. The method of Claim 8, wherein said plurality of reduced threat signature sets comprises a set of signatures common to more than one operating system type.

10. The method of Claim 8, wherein said fingerprinting comprises active fingerprinting.

11. The method of Claim 8, wherein said fingerprinting comprises passive fingerprinting.

12. The method of Claim 8, wherein said target operating system type is the packet source host operating system.

13. The method of Claim 8, wherein said target operating system type is the packet destination host operating system.

14. The method of Claim 8, wherein said target OS type is both the packet source host OS and the packet destination host OS.

15. The method of Claim 8, wherein, when said fingerprinting identifies more than one target operating system type, said directing to the virtual IDS process corresponding to each said target operating system type occurs at substantially the same time.

16. An apparatus for intrusion detection system (IDS) virtualization, comprising:

a first network interface connected to an unprotected network;

a fingerprinter operably connected to said first network interface for receiving a plurality of packets and configured to determine at least one corresponding target operating system (OS) fingerprint for each said packet;

a director connected to said fingerprinter configured to receive said plurality of packets and said corresponding target OS fingerprints, wherein said director directs each said packet to one or more virtual IDS units according to said at least one corresponding OS fingerprint; and

a second network interface connecting said one or more virtual IDS units to a protected network;

wherein at least one said virtual IDS units comprises at least one threat signature specific to an operating system and is configured to accept only packets that do not match any said threat signature.

17. The apparatus of Claim 16, wherein said fingerprinter employs passive fingerprinting algorithms.

18. The apparatus of Claim 16, wherein said fingerprinter employs active fingerprinting algorithms.

19. The apparatus of Claim 16, wherein, when said fingerprinter identifies more than one target OS fingerprint, said director directs each said packet to said virtual IDS units at substantially the same time.

20. An apparatus for intrusion detection system virtualization, comprising:

means for receiving a stream of packets;

means for fingerprinting each packet in a said stream to identify at least one target operating system (OS) type;

means for directing each said packet to a virtual IDS process corresponding to each said identified target OS type;

means for comparing each said packet to a threat signature set corresponding to each said identified target OS type in said virtual IDS process; and

means for accepting each said packet based on said comparing.

21. The apparatus of Claim 20, wherein said means for fingerprinting comprise means for active fingerprinting.

22. The apparatus of Claim 20, wherein said means for fingerprinting comprise means for passive fingerprinting.

23. The apparatus of Claim 20, wherein said target OS type is the packet source host OS.

24. The apparatus of Claim 20, wherein said target OS type is the packet destination host OS.

25. The apparatus of Claim 20, wherein said target OS type is both the packet source host OS and the packet destination host OS.

26. The apparatus of Claim 20, wherein, when said means for fingerprinting identifies more than one target OS type, said means for directing operates at substantially the same time.

27. A computer system for use in intrusion detection system virtualization, comprising computer instructions for:

receiving a stream of packets;

fingerprinting each packet in a said stream to identify at least one target operating system (OS) type;

directing each said packet to a virtual IDS process corresponding to each said identified target OS type;

comparing each said packet to a threat signature set corresponding to each said identified target OS type in said virtual IDS process; and

accepting each said packet based on said comparing.

28. The computer system of Claim 27, wherein said computer instructions for fingerprinting comprise computer instructions for active fingerprinting.

29. The computer system of Claim 27, wherein said computer instructions for fingerprinting comprise computer instructions for passive fingerprinting.

30. A computer-readable medium storing a computer program executable by a plurality of server computers, the computer program comprising computer instructions for:

receiving a stream of packets;

fingerprinting each packet in a said stream to identify at least one target operating system (OS) type;

directing each said packet to a virtual IDS process corresponding to each said identified target OS type;

comparing each said packet to a threat signature set corresponding to each said identified target OS type in said virtual IDS process; and

accepting each said packet based on said comparing.

31. The computer-readable medium of Claim 30, wherein said computer instructions for fingerprinting comprise computer instructions for active fingerprinting.

32. The computer-readable medium of Claim 30, wherein said computer instructions for fingerprinting comprise computer instructions for passive fingerprinting.

33. A computer data signal embodied in a carrier wave, comprising computer instructions for:

receiving a stream of packets;

fingerprinting each packet in a said stream to identify at least one target operating system (OS) type;

directing each said packet to a virtual IDS process corresponding to each said identified target OS type;

comparing each said packet to a threat signature set corresponding to each said identified target OS type in said virtual IDS process; and

accepting each said packet based on said comparing.

34. The computer data signal of Claim 33, wherein said computer instructions for fingerprinting comprise computer instructions for active fingerprinting.

35. The computer data signal of Claim 33, wherein said computer instructions for fingerprinting comprise computer instructions for passive fingerprinting.

Fig. 1

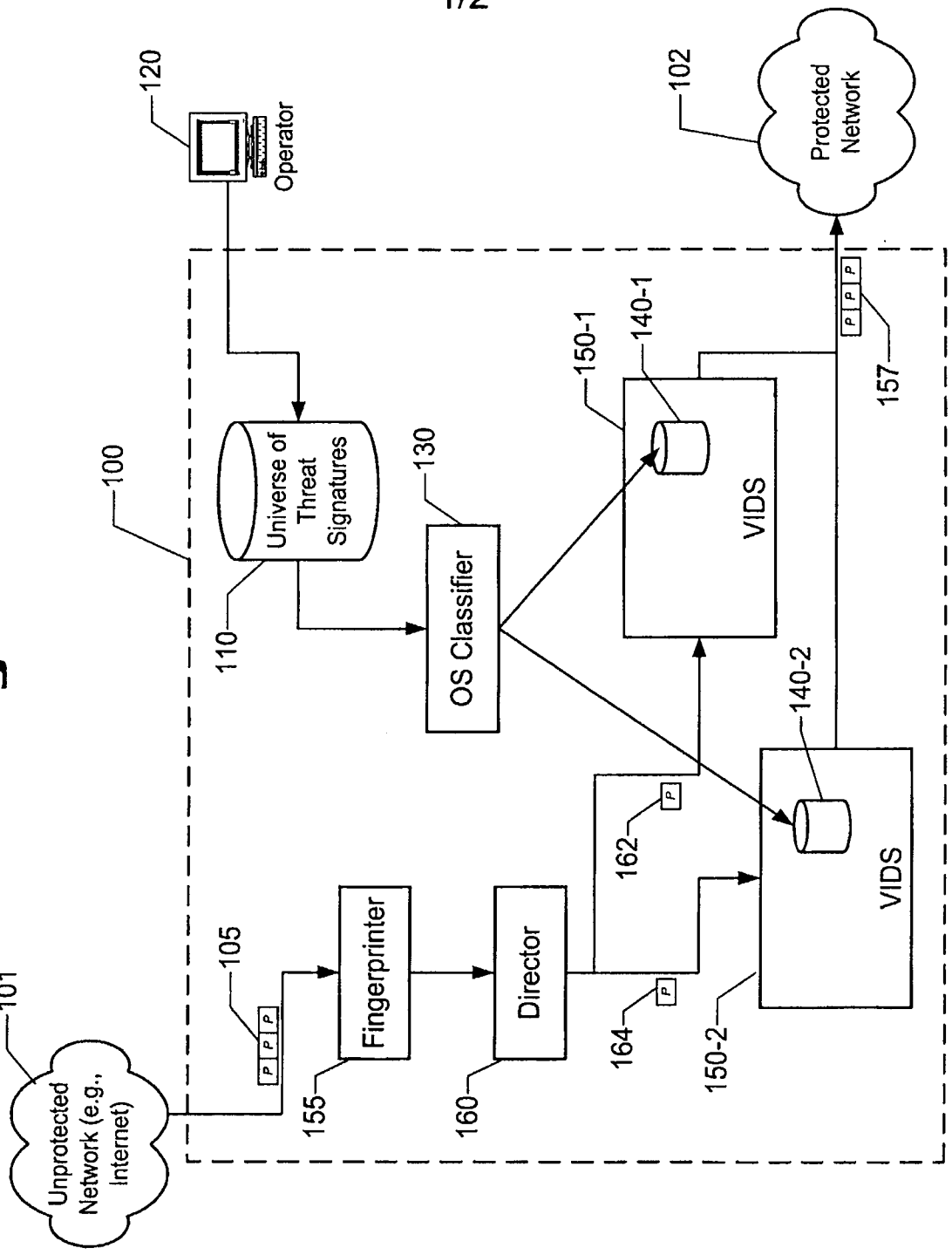
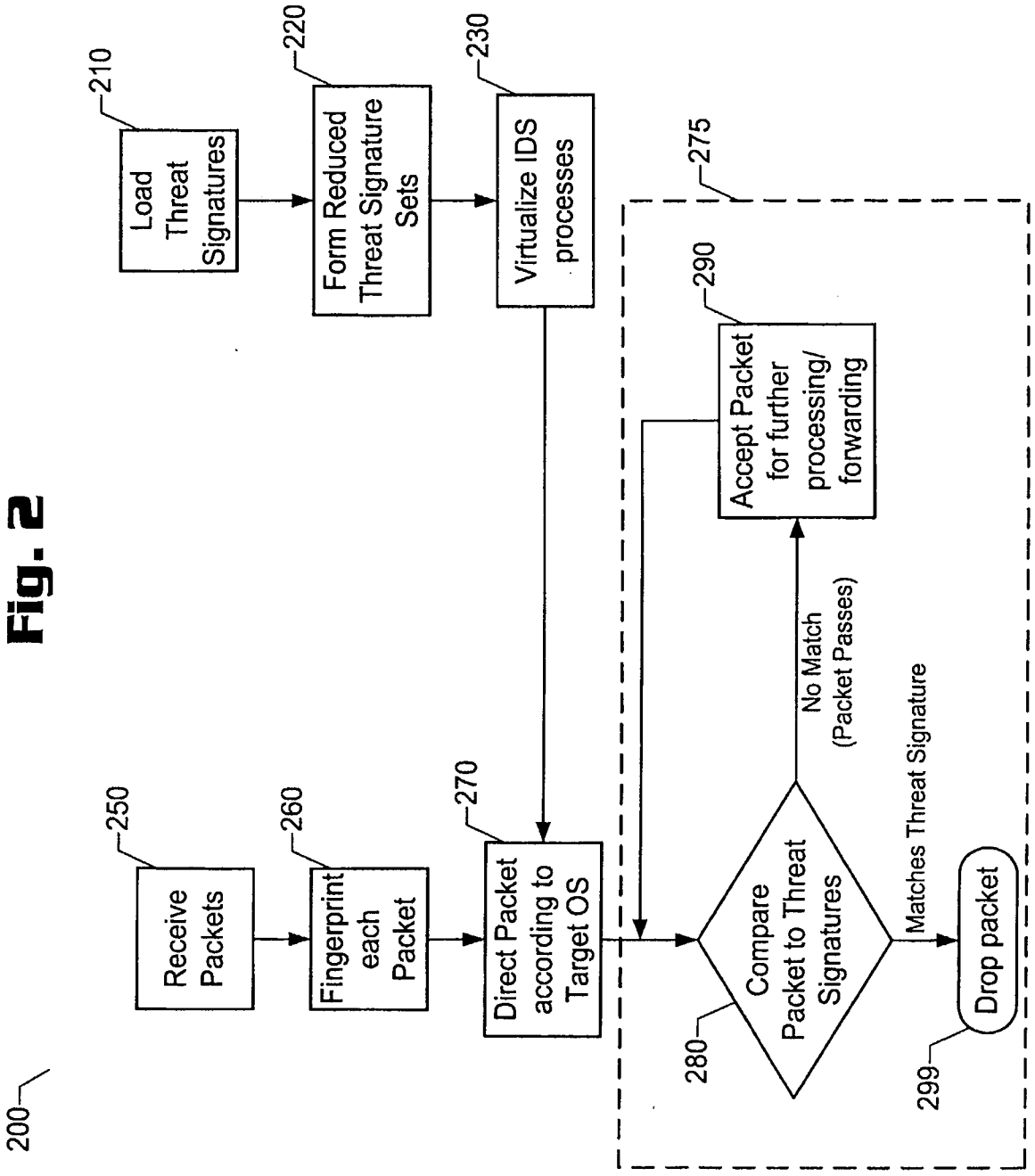

Unprotected Network (e.g., Internet)
101
Operator
120
Protected Network
102
100
Universe of Threat Signatures
110
OS Classifier
130
VIDS
150-1
140-1
VIDS
140-2
150-2
Fingerprinter
155
Director
160
105
162
164
157
P

Fig. 2


200
210
Load Threat Signatures
220
Form Reduced Threat Signature Sets
230
Virtualize IDS processes
275
250
Receive Packets
260
Fingerprint each Packet
270
Direct Packet according to Target OS
290
Accept Packet for further processing/ forwarding
280
Compare Packet to Threat Signatures
No Match (Packet Passes)
Matches Threat Signature
299
Drop packet

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US2005/011702

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04L29/06

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, IBM-TDB, INSPEC, WPI Data, PAJ

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2003/212910 A1 (ROWLAND CRAIG H ET AL) 13 November 2003 (2003-11-13) figure 3 paragraph [0006] paragraph [0024] paragraph [0031] - paragraph [0032] -----	1-35
E,X	US 2005/086522 A1 (ROWLAND CRAIG H) 21 April 2005 (2005-04-21) figure 3 paragraph [0004] paragraph [0028] - paragraph [0029] -----	1-35

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
6 July 2005	09. 09. 2005

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Raposo Pires, J

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US2005/011702

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	DEBAR H ET AL: "Towards a taxonomy of intrusion-detection systems" COMPUTER NETWORKS, ELSEVIER SCIENCE PUBLISHERS B.V., AMSTERDAM, NL, vol. 31, no. 8, 23 April 1999 (1999-04-23), pages 805-822, XP004304519 ISSN: 1389-1286 page 808, paragraph 3.1.1. page 809, paragraph 3.1.1.2 page 817, paragraph 4	1-35
A	US 2003/188189 A1 (DESAI ANISH P ET AL) 2 October 2003 (2003-10-02) paragraph [0035] paragraph [0038] paragraph [0048]	1-35

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/US2005/011702

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2003212910 A1	13-11-2003	US 2003196123 A1	16-10-2003
		AU 2003220582 A1	13-10-2003
		CA 2479504 A1	09-10-2003
		EP 1491019 A1	29-12-2004
		WO 03084181 A1	09-10-2003
US 2005086522 A1	21-04-2005	WO 2005041141 A2	06-05-2005
US 2003188189 A1	02-10-2003	NONE	

DERWENT-ACC-NO: 2005-811408

DERWENT-WEEK: 200757

TITLE: Virtualization method for network intrusion detection system for monitoring network traffic, involves comparing each packet to threat signature set corresponding to target operating system type identified by active fingerprinting

INVENTOR: ITHAL R; ITHAL R G

PATENT-ASSIGNEE: CISCO TECH IND[CISCN] , CISCO TECHNOLOGY INC[CISCN]

PRIORITY-DATA: 2004US-832588 (April 27, 2004)

PATENT-FAMILY:

PUB-NO	PUB-DATE	LANGUAGE
WO 2005109824 A1	November 17, 2005	EN
EP 1741265 A1	January 10, 2007	EN
CN 1943210 A	April 4, 2007	ZH

DESIGNATED-STATES: AE AG AL AM AT AU AZ BA BB BG BR BW BY BZ CA CH CN CO CR CU CZ DE DK DM DZ EC EE EG ES FI GB GD GE GH GM HR HU ID IL IN IS JP KE KG KM KP KR KZ LC LK LR LS LT LU LV MA MD MG MK MN MW MX MZ NA NI NO NZ OM PG PH P L PT RO RU SC SD SE SG SK SL SM SY TJ TM TN TR TT TZ UA UG US UZ VC VN YU ZA ZM ZW AT BE BG BW CH CY CZ DE DK EA EE ES FI FR GB GH GM GR HU IE IS IT KE LS LT LU MC MW MZ NA NL OA PL PT RO SD SE SI SK SL SZ TR TZ UG ZM ZW AT BE BG CH CY CZ DE DK EE ES FI F R GB GR HU IE IS IT LI LT LU MC NL PL PT RO SE SI SK TR

APPLICATION-DATA:

PUB-NO	APPL-DESCRIPTOR	APPL-NO	APPL-DATE
WO2005109824A1	N/A	2005WO-US011702	April 5, 2005
CN 1943210A	N/A	2005CN-80011926	April 5, 2005
EP 1741265A1	N/A	2005EP-736313	April 5, 2005
EP 1741265A1	Based on	2005WO-US011702	April 5, 2005

INT-CL-CURRENT:

TYPE	IPC DATE
CIPP	H04L29/06 20060101
CIPP	H04L29/06 20060101

CIPS H04L29/06 20060101

ABSTRACTED-PUB-NO: WO 2005109824 A1

BASIC-ABSTRACT:

NOVELTY - The method involves performing active fingerprinting of each packet in a received stream (105) to identify a target operating system (OS) type. Each packet is directed to a virtual intrusion detection system (IDS) corresponding to each identified target OS type. Each packet is compared to threat signature sets (140-1,140-2) corresponding to each identified OS type and packet is accepted accordingly.

DESCRIPTION - INDEPENDENT CLAIMS are also included for the following:

(1) virtualization apparatus;

(2) computer system for use in network intrusion detection system virtualization;

(3) computer readable medium storing network intrusion detection system virtualization program; and

(4) computer data signal for network intrusion detection system virtualization.

USE - For virtualizing network intrusion detection system (IDS) for monitoring network traffic and for blocking attempted attacks on or intrusion into protected network space e.g. for local area network (LAN), metropolitan area network (MAN), etc.

ADVANTAGE - Enhances the processing efficiency and speed by reducing the set of threat signatures that need to be searched for each packet.

DESCRIPTION OF DRAWING(S) - The figure shows a high level block diagram of the virtualized intrusion detection system (VIDS).

virtualized intrusion detection system (100)

stream of packets (105)

threat signature sets (140-1,140-2)

VIDS (150-1,150-2)

finger printer (155)

CHOSEN-DRAWING: Dwg.1/2

TITLE-TERMS: METHOD NETWORK INTRUDE DETECT SYSTEM MONITOR TRAFFIC COMPARE PACKET THREAT SIGNATURE SET CORRESPOND TARGET OPERATE TYPE IDENTIFY ACTIVE FINGERPRINT

DERWENT-CLASS: T01 W01

EPI-CODES: T01-N02B2B; T01-N02B2C; T01-S03; W01-A06A3;

SECONDARY-ACC-NO:

Non-CPI Secondary Accession Numbers: 2005-672792